Test Instructions

Target stack:
React 19.2, Next.js 16.1, TypeScript 5.9, Tailwind CSS 4.1, CSS Grid and Flexbox

Goal

Build a single production quality mobile responsive Product Details flow that matches the Figma UI provided. Use Next.js API Routes for the minimal backend surface. This will evaluate React, Next.js, TypeScript, Tailwind, layout skills, and simple server integration.

Visual reference

Use the exact UI reference. Match layout, spacing, typography, and component behavior as closely as possible.

Required deliverables

1. Public GitHub repo with meaningful commits.
2. README with run steps, build steps, and deployment instructions. Include any decisions or shortcuts.
3. A working app locally. Deployment to Vercel or similar is optional but encouraged.
4. Short demo screencast 1 to 3 minutes showing the product page, add to cart, and order create flow.

Required features (minimal)

Frontend

- Product detail screen (mobile) matching the provided UI, including: product image, rounded corners, size selector, color swatches, price, add to cart.
- Responsive layout that looks correct on small screens.
- Persist selected size and color in cart items.

API (Next.js API Routes)

- GET /api/products — returns product list or seeded product JSON
- GET /api/products/:id — product detail
- POST /api/cart — add item to session backed cart or in memory store
- POST /api/orders — create order from cart, simulate payment success and return order id

Notes on backend and services

- You may use an in memory store, a local JSON file, or a light DB. In memory or file based storage is acceptable for passing.
- Protect cart and order routes with a simple session or mock auth. Full OAuth is not required. A seeded test user is fine.
- Images may be served from public folder, repo, or any CDN.

What we must see in the submission

- UI visually matches the provided product details screen and is responsive.
- Add to cart and create order flows work end to end via the API routes.
- Code is typed with TypeScript for main models.
- README explains how to run locally and how to seed data. Include env.example for any env vars you use.